

Exponate in der IT Sammlung – Museumsführer HoloMu

HoloLens

Als Plattform zur Informationsvermittlung wird Microsofts HoloLens auf Basis von Unity3D und C# verwendet. Durch diese Augmented-Reality-Brille kann der Nutzer seine Umgebung und somit auch das Museum und dessen Exponate ungestört wahrnehmen. Möchte der Nutzer mehr über ein Exponat erfahren, kann er durch Benutzung der Bilderkennung zunächst allgemeine, auf Wunsch auch weiterführende, Informationen einblenden lassen. Die Bilderkennung kann über drei verschiedene Methoden ausgelöst werden. Möchte der Nutzer auf eine manuelle Lösung setzen, ist es ihm möglich entweder den mit der HoloLens mitgelieferten „Klicker“ zu betätigen oder die „Klick“-Geste ausführen. Außerdem wird eine automatische Auslösung implementiert. Wenn sich das Blickfeld des Nutzers für zwei Sekunden nicht bzw. nur unwesentlich verändert, wird ebenfalls die Bilderkennung gestartet. Während dieser zwei Sekunden, erhält der Nutzer über einen sich füllenden Kreis Feedback über den Status der automatischen Erkennung. In allen drei Fällen wird ein Bild vom aktuellen Blickfeld des Nutzers abgegriffen und über eine zu implementierende Schnittstelle an die API gesendet (siehe unten).

Die erhaltenen Exponat-Informationen werden dem Nutzer in einer stilistisch ansprechenden Textbox präsentiert. Diese ist zunächst an die Bewegungen des Nutzers gekoppelt. Führt der Nutzer die „Klick“-Geste auf der Textbox aus, wird sie an der aktuellen Position im Raum fixiert. Dies ist das Standardverhalten von Fenstern in der HoloLens. Recommender-Informationen werden in einem kleinen Textfeld in einer Ecke des Blickfelds angezeigt. Über nicht erfolgreiche Bilderkennungsversuche wird der Nutzer nur informiert, wenn er die Bilderkennung manuell gestartet hat, oder wenn die ermittelte Wahrscheinlichkeit in einem noch zu bestimmenden Bereich liegt. Letzteres wird serverseitig entschieden und an späterer Stelle genauer erläutert. Da die Exponat-Informationen bereits aufbereitet an die HoloLens geschickt werden, kann die Aufteilung in Kategorien problemlos erfolgen. Diese sind über entsprechende Buttons zugänglich.

Wird ein Exponat erfolgreich erkannt, wird die aktuelle Position des Nutzers gespeichert. Verlässt der Nutzer den Bereich des Exponats, wird das Textfeld ausgeblendet und durch ein auffälliges Icon ersetzt. Betritt der Nutzer zu einem späteren Zeitpunkt wieder den Bereich, öffnet sich auf Wunsch des Nutzers wieder das Info-Feld, ohne dass wiederum die Bilderkennung gestartet werden muss.

REST API

Hosting

Um den Prototypen praxisnah umzusetzen, werden die Bilderkennung, die Datenspeicherung und das Recommendersystem auf einem Server ausgelagert. Dadurch besteht die Möglichkeit, später weitere Exponate mit Informationen auszustatten, ohne dass etwas an dem auf der HoloLens installierten Programm geändert werden muss. Sollte **HoloMu** in einem Museum auf mehreren HoloLens's zum Einsatz kommen, wäre es außerdem höchst unpraktisch, wenn auf allen Geräten die Trainingsdaten hinterlegt werden müssten.

Die API wird in Python unter Verwendung der Bibliothek *flask* implementiert. Gehostet wird sie entweder einem Server der Universität, oder, wenn dies scheitert, auf einem Server von *Uberspace*.

Bilderkennung

Die Bilderkennung erfolgt unter Benutzung des Frameworks *TensorFlow*. Um den Bilderkenner zu trainieren, werden von jedem Exponat circa 100 Fotos aus verschiedenen Winkeln und Abständen gemacht. Die Bilder liegen strukturiert auf dem Server und werden bei Serverstart eingelesen und zum Training benutzt. Die API erhält einen Endpunkt, welcher ein Bild empfangen kann. Dieses Bild wird zunächst auf dem Server gespeichert und dann vom Bilderkenner untersucht. Das Ergebnis wird anhand der Wahrscheinlichkeit einer korrekten Erkennung in drei Kategorien eingeteilt. Ist die Erkennungswahrscheinlichkeit sehr hoch, werden die Informationen für das erkannte Exponat aus dem Datensatz geladen und als Antwort zurück an die HoloLens gesendet. Ist nicht sicher, ob eine korrekte Erkennung stattgefunden hat, wird eine Fehlermeldung zurückgegeben. Kann das System davon ausgehen, dass kein Exponat erkannt wurde, wird nichts zurückgegeben. Durch diese Abstufung wird erreicht, dass der Nutzer einerseits bei falscher fehlerhafter Erkennung sofort Feedback erhält und nochmals die Erkennung starten kann und andererseits nicht dauerhaft Fehlermeldungen erhält, obwohl er die Bilderkennung nicht absichtlich ausgelöst hat. Letzteres kann bei der oben beschriebenen automatischen Auslösung durchaus passieren.

Ist die Erkennung abgeschlossen, wird das hochgeladene Bild wieder gelöscht. Außerdem wird bei erfolgreicher Erkennung die ID des erkannten Objekts in der *MySQL* Datenbank abgespeichert, um später eine Empfehlung aussprechen zu können.

Datenaufbereitung

Zur Bilderkennung im Kontext der IT-Sammlung werden neben den 100 Fotos pro Exponat noch die vorhandenen Informationen der bereits ausgestellten Informationstafeln, auch relevante Informationen aus einer vorhandenen Datenbank begutachtet und aufbereitet. Durch die Umstrukturierung der vorhandenen Daten ergibt sich die Möglichkeit die Informationen in vordefinierte Interessensgebiete einzuordnen. Der Datensatz an sich wird als XML Dokument auf dem Server hinterlegt. Hierfür wird einer der beiden oben genannten Hosting-Provider verwendet. Somit ist sichergestellt, dass auf dem Server sowohl für die Fotos als auch den Datensatz genügend Speicher zur Verfügung steht.

Zudem soll noch wie bereits erwähnt ein Recommendersystem implementiert werden. Das setzt voraus, dass die Objekte entsprechend ausgewählt werden, um sie in Kategorien unterteilen und korrespondierend im XML-Dokument auszeichnen zu können. Dieses Dokument untergliedert sich in die einzelnen Objekte, die mittels einer eindeutigen ID identifizierbar sind. Jedes Objekt erhält mehrere Elemente, wie zum Beispiel Name, Kategorie, Jahr, Hersteller und eine Kurzbeschreibung. Weitere Elemente, um mehr Informationen zu einem Objekt zu erhalten, werden mittels Attributen ausgezeichnet und anhand des Wertes spezifiziert.

Recommendersystem

Der Recommender soll serverseitig implementiert werden. Dabei ist zunächst nur angedacht auf den aktuellen Nutzer zu reagieren, um diesem, ausgehend von seinen eigenen bisherigen erfassten Daten,

Vorschläge unterbreiten zu können. Um den Recommender sinnvoll einzusetzen und dem Nutzer keine irreführenden Informationen zu übermitteln, wird die Empfehlung für ein weiteres Objekt erst nach einer gewissen Anzahl an Objekten geliefert. Diese wird im Verlauf der Implementierung bzw. in einer Testing-Phase genauer bestimmt werden können. Der Recommender soll ausgelöst werden, wenn der Nutzer den Bereich eines Exponates verlässt, um den Museumsbesucher nicht während einer Objekterkennung oder dem Lesen von Objektinformationen abzulenken. Die Empfehlung selbst soll auf Wahrscheinlichkeiten basieren und wird gegebenenfalls aus einem gewichteten Durchschnittswert der Betrachtungsreihenfolge oder der Betrachtungsdauer gebildet. Um zu verhindern, dass der Benutzer aufgrund der zuerst betrachteten Objekte nur noch Exponate dieser Kategorie als Vorschlag erhält, ordnen wir den Kategorien zunächst eine bestimmte relative Häufigkeit zu. Dies ermöglichtes die relativen Häufigkeiten der Kategorien mit der aktuell betrachteten Kategorie jeweils zu verrechnen und daher mit höherer Wahrscheinlichkeit ein Exponat aus der bevorzugten Kategorie als Vorschlag zu erhalten. Zudem ist jedoch sichergestellt, dass auch Exponate aus einer anderen Kategorie berücksichtigt werden.